**CryptCloud+ Secure and Expressive Data Access Control for Cloud Storage**

CryptCloud Secure and Expressive Data Access Control for Cloud Storage

A Project Report submitted in partial fulfillment of the degree of the   
 Bachelor of Technology in Computer Science and Engineering

By

Student names Roll Numbers

MD.ABRAR - 19C41A05F6

L.CHAITHANYA - 19C41A05D8

N. SAIRAM - 19C41A05G0

Under the Guidance of

**Mrs. G.Gouthami**



**Department of Computer Science and Engineering**

**Jayamukhi institute of technological Sciences**

**Narsampet, warangal-506 332**

**(Affiliated to JNTUH, Accredited by NAAC ‘A’ Grade)**

**2023**

**Jayamukhi institute of technological Sciences**

**Narsampet, warangal – 506 332**

**(Affiliated to JNTUH, Accredited by NAAC ‘A’ Grade)**

**CERTIFICATE**



This is to certify that the Project Report entitled “**CRYPTCLOUD+ SECURE AND EXPRESSIVE DATA ACCESS CONTOL FOR CLOUD STORAGE**” is a bona fide work of the students **MD.ABRAR,L.CHAITANYA,N.SAIRAM** bearing Roll No.s **19C41A05F6,19C41A05D8 ,19C41A05G0** submitted in partial fulfillment of the requirements for the award of the degree of ***Bachelor of Technology*** in **Computer Science & Engineering** during the academic year **2022-23.**

**Guide Head of the Department**

**Principal**

**ACKNOWLEDGEMENT**

We would especially like to express our extreme gratitude and sincere thanks to our guide **Mrs. G.Gouthami ,Asst. Prof**., Department of Computer Science and Engineering for her enthusiastic and innovative guidance and support.

We are extremely thankful to **Dr. P. Srinivas Rao**, Head, Department of Computer Science and Engineering for the support and encouragement accorded to carry out this project work.It gives us immense pleasure in expressing sincere and deep sense of gratitude to **Prof.V. Janaki**, Principal, Jayamukhi Institute of Technological sciences for the facility made available for the progress and completion of our work.

We are also thankful to our management for providing all the facilities for completing project.

**ABSTRACT**

Secure cloud storage, which is an emerging cloud service, is designed to protect the confidentiality of outsourced data but also to provide flexible data access for cloud users whose data is out of physical control. Ciphertext-Policy Attribute-Based Encryption (CP-ABE) is regarded as one of the most promising techniques that may be leveraged to secure the guarantee of the service. However, the use of CP-ABE may yield an inevitable security breach which is known as the misuse of access credential (i.e. decryption rights), due to the intrinsic “all-or-nothing” decryption feature of CP-ABE. In this paper, we investigate the two main cases of access credential misuse: one is on the semi-trusted authority side, and the other is on the side of cloud user. To mitigate the misuse, we propose the first accountable authority and revocable CP-ABE based cloud storage system with white-box traceability and auditing, referred to as CryptCloud+. We also present the security analysis and further demonstrate the utility of our system via experiments.

**CONTENTS**

**TOPIC PAGE NO**

INTRODUCTION 1-4

LITERATURE REVIEW 5-10

SYSTEM REQUIREMENT 11

DATA FLOW DIAGRAM 12

ARCHITECTURE DIAGRAM 13

FLOW CHART : DATA OWNER 14

FLOW CHART : DATA USER 15

FLOW CHART : PUBLIC CLOUD 16

FLOW CHART : AUDITOR & STA 17

CLASS DIAGRAM 18

SEQUENCE DIAGRAM 19

IMPLEMENTATION 20

SYSTEM TESTING 21-31

Testing Methodologies

User Training

Testing Strategy

Preliminary Investigation

Request Clarification

Feasibility Analysis

Request Approval

SYSTEM DESIGN AND DEVELOPMENT 32-33

Input Design

Out Design

CONCLUSION AND FUTURE WORK 34

BIBLIOGRAPHY 35-37

INTRODUCTION

THE prevalence of cloud computing may indirectly incur vulnerability to the conﬁdentiality of outsourced data and the privacy of cloud users. A particular challenge here is on how to guarantee that only authorized users can gain access to the data, which has been outsourced to cloud, at anywhere and anytime . One naive solution is to employ encryption technique on the data prior to uploading to cloud. However, the solution limits further data sharing and processing. This is so because a data owner needs to download the encrypted data from cloud and further re-encrypt them for sharing (suppose the data owner has no local copies of the data). A ﬁne-grained access control over encrypted data is desirable in the context of cloud computing . Ciphertext-Policy Attribute-Based Encryption (CPABE) may be an effective solution to guarantee the conﬁdentiality of data and provide ﬁne-grained access control here. In a CP-ABE based cloud storage system, for example, organizations (e.g., a university such as the UniversityofTexasatSanAntonio)andindividuals(e.g.,students,facultymembers and visiting scholars of the university) can ﬁrst specify access policyoverattributesofapotentialclouduser.Authorizedcloudusersthenaregrantedaccesscredentials (i.e., decryption keys) corresponding to their attribute sets (e.g., student role, faculty member role, or visitor role), which can be used to obtain access to the outsourced data. As a robust one-to-many encryption mechanism, CP-ABE offers a reliable method to protect data stored in cloud, but also enables ﬁne-grained access control over the data. Generally speaking, the existing CP-ABE based cloud storage systems fail to consider the case where access credential is misused. For instance, a university deploys a CPABE based cloud storage system to outsource encrypted student data to cloud under some access policies that are compliant with the relevant data sharing and privacy legislation (e.g., the federal Family Educational Rights and Privacy Act (FERPA) and Health Insurance Portability and Accountability Act of 1992 (HIPAA)). The ofﬁcial in charge at the organization (e.g. university’s security manager) initializes the system parameters and issues access credentials for all users (e.g., students, faculty members, and visiting scholars). Each employee is assigned with several attributes (e.g.,“administrator”,“seniormanager”,“ﬁnancialofﬁcer”, “tenured faculty”, “tenure-track faculty”, “non tenure-track faculty”, “instructors”, “adjunct”, “visitor”, and/or “students”). Only the employees with attributes satisfying the decryption policy of the outsourced data are able to gain access to the student data stored in cloud (e.g. student admission materials). As we may have known, the leakage of any sensitive student information stored in cloud could result in a range of consequences for the organization and individuals (e.g., litigation, loss of competitive advantage, and criminal charges). The CP-ABE may help us prevent security breach from outside attackers. But when an insider of the organization is suspected to commit the “crimes” related to the redistribution of decryption rights and the circulation of student information in plainformat forillicit ﬁnancial Isit also possible for us to revoke the compromised access privileges? In addition to the above questions, we have one more which is related to key generation authority. A cloud user’s access credential (i.e., decryption key) is usually issued by a semi-trusted authority based on the attributes the user possesses. How could we guarantee that this particular authority will not (re-)distribute the generated access credentials to others? For example, the organization security ofﬁcial leaks a lecturer Alice’s key to an outsider Bob (who is not the employee of the university). One potential answer to the question is to employ multiple authorities. Nevertheless, this incurs additional cost in communication and infrastructure deployment and meanwhile, the problem of malicious collusion among authorities remains. Therefore, we posit that adopting an accountable authority approach to mitigate the access credential escrow problem is the preferred strategy. Seeking to mitigate access credential misuse,we propose CryptCloud+, an accountable authority and revocable CPABE based cloud storage system with white-box traceability and auditing. To the best of our knowledge, this is the ﬁrst practical solution to secure ﬁne-grained access control over encrypted data in cloud. Speciﬁcally, in our work, we ﬁrst present a CP-ABE based cloud storage framework. Using this (generic) framework, we propose two accountable authority and revocable CP-ABE systems (with whitebox traceability and auditing) that are fully secure in the standardmodel,referredtoasATER-CP-ABEandATIR-CPABE, respectively. Based on the two systems, we present the construction of CryptCloud+ that provides the following features.

1) Traceability of malicious cloud users. Users who leak their access credentials can be traced and identiﬁed.

2) Accountable authority. A semi-trusted authority, who (without proper authorization) generates and further distributes access credentials to unauthorized user(s), can be identiﬁed. This allows further actions to be undertaken (e.g. criminal investigation or civil litigation for damages and breach of contract).

3) Auditing. An auditor can determine if a (suspected) cloud user is guilty in leaking his/her access credential.

4) “Almost” zero storage requirement for tracing. We use a Paillier-likeencryptionasanextractablecommitmentintracingmaliciouscloudusersandmorepractically, we do not need to maintain an identity table of users for tracing (unlike the approach used in .

5) Malicious cloud user’s revocation. Access credentials for individual traced and further determined to be “compromised” can be revoked. We design two mechanisms to revoke the “traitor(s)” effectively. The ATER-CP-ABE provides an explicitly revocation mechanism where a revocation list is speciﬁed explicitly into the algorithm Encrypt, while the ATIRCP-ABE offers an implicitly revocation where the encryption does not need to know the revocation list but a key update operation is required periodically. This paper extends our earlier work (a conference version in as follows.

1) We present a formal framework model of the proposed system, designed for practical cloud storage system deployment.

2) We address a weakness in the auditing procedure of the conference version. Speciﬁcally, a malicious user may change tid of his secret key in the conference version, and the auditing procedure will fail in this case. As a mitigation, we revise the key generation algorithm and add an audit list to detect if the tid is changed.

3) We enhance the functionality of the construction (w.r.t. AAT-CP-ABE) proposed in the conference version and further present two enhanced constructions, namely ATER-CP-ABE and ATIR-CP-ABE. These constructions allow us to effectively revoke the malicious users explicitly or implicitly. We also present the new deﬁnitions, technique and related materials of ATER-CP-ABE and ATIR-CP-ABE.

4) Based on the new ATER-CP-ABE and ATIR-CPABE, we present CryptCloud+ which is an effective and practical solution for secure cloud storage.

5) We provide general extensions (of our system) on the large universe, the multi-use, and the prime-order setting cases, so that the solution introduced in this paper is more scalable in real-world applications.

6) We comprehensively evaluate the efﬁciency of the proposed ATER-CP-ABE and ATIR-CP-ABE via experiments.

**LITERATURE REVIEW**

**L1** : **Secure and Privacy Preserving Protocol for Cloud-Based Vehicular DTN’s**

**Authors**:[**Xiaolei Dong**](https://www.researchgate.net/scientific-contributions/Xiaolei-Dong-26819504)**, Xiaolei Dong,Zhenfu Cao, Athanasios Vasilakos**

**Publlished year: 2015.**

**Description:**Cloud-assisted vehicular delay tolerant networks (DTNs) have been utilized in wide-ranging applications where a continuous end-to-end connection is unavailable, the message transmission is fulfilled by the cooperation among vehicular nodes and follows a store-carry-and-forward manner, and the complex computational work can be delegated to the disengaged vehicles in the parking lots which constitute the potential vehicular cloud. Nevertheless, the existing incentive schemes as well as the packet forwarding protocols cannot well model continuous vehicle collaboration, resist vehicle compromise attacks and collusion attacks, leaving the privacy preservation issues untouched. In this paper, a novel threshold credit-based incentive mechanism (TCBI) is proposed based on the modified model of population dynamics to efficiently resist the node compromise attacks, stimulate the cooperation among intermediate nodes, maximize vehicular nodes' interest, and realize the fairness of possessing the same opportunity of transmitting packets for credits. Then, a TCBI-based privacy-preserving packet forwarding protocol is proposed to solve the open problem of resisting layer-adding attack by outsourcing the privacy-preserving aggregated transmission evidence generation for multiple resource-constrained vehicles to the cloud side from performing any one-way trapdoor function only once. The vehicle privacy is well protected from both the cloud and transportation manager. Finally, formal security proof and the extensive simulation show the effectiveness of our proposed TCBI in resisting the sophisticated attacks and the efficiency in terms of high reliability, high delivery ratio, and low average delay in cloud-assisted vehicular DTNs.

# L2: Security and Privacy for Cloud-Based IoT: Challenges

**Authors:** Zhenfu Cao, Jun Zhou, Xiaolei Dong, Athanasios v.vasilakos.

**Published year** : 2019.

**Description**:The Internet of Things is increasingly becoming a ubiquitous computing service, requiring huge volumes of data storage and processing. Unfortunately, due to the unique characteristics of resource constraints, self-organization, and shortrange communication in IoT, it always resorts to the cloud for outsourced storage and computation, which has brought about a series of new challenging security and privacy threats. In this article, we introduce the architecture and unique security and privacy requirements for the next generation mobile technologies on cloud-based IoT, identify the inappropriateness of most existing work, and address the challenging issues of secure packet forwarding and efficient privacy preserving authentication by proposing new efficient privacy preserving data aggregation without public key homomorphic encryption. Finally, several interesting open problems are suggested with promising ideas to trigger more research efforts in this emerging area.

**L3: Outsourced Attribute-Based Encryption with Keyword Search Function for Cloud Storage**

**Authors:** Jiguo Li ,Xiaonan Li ,Yichen Zhang,Jinguang Han**.**

**Published year: 2017**

**Description:** Cloud computing becomes increasingly popular for data owners to outsource their data to public cloud servers while allowing intended data users to retrieve these data stored in cloud. This kind of computing model brings challenges to the security and privacy of data stored in cloud. Attribute-based encryption (ABE) technology has been used to design fine-grained access control system, which provides one good method to solve the security issues in cloud setting. However, the computation cost and ciphertext size in most ABE schemes grow with the complexity of the access policy. Outsourced ABE (OABE) with fine-grained access control system can largely reduce the computation cost for users who want to access encrypted data stored in cloud by outsourcing the heavy computation to cloud service provider (CSP). However, as the amount of encrypted files stored in cloud is becoming very huge, which will hinder efficient query processing. To deal with above problem, we present a new cryptographic primitive called attribute-based encryption scheme with outsourcing key-issuing and outsourcing decryption, which can implement keyword search function (KSF-OABE). The proposed KSF-OABE scheme is proved secure against chosen-plaintext attack (CPA). CSP performs partial decryption task delegated by data user without knowing anything about the plaintext. Moreover, the CSP can perform encrypted keyword search without knowing anything about the keywords embedded in trapdoor.

**L4** : **Flexible and Fine-Grained Attribute-Based Data Storage in Cloud Computing**

**Authors**: Jiguo Li,Wei Yao,Yichen Zhang, Huiling Qian,Jinguang Han

**Published year:2017**

**Description:**  With the development of cloud computing, outsourcing data to cloud server attracts lots of attentions. To guarantee the security and achieve flexibly fine-grained file access control, attribute based encryption (ABE) was proposed and used in cloud storage system. However, user revocation is the primary issue in ABE schemes. In this article, we provide a ciphertext-policy attribute based encryption (CP-ABE) scheme with efficient user revocation for cloud storage system. The issue of user revocation can be solved efficiently by introducing the concept of user group. When any user leaves, the group manager will update users' private keys except for those who have been revoked. Additionally, CP-ABE scheme has heavy computation cost, as it grows linearly with the complexity for the access structure. To reduce the computation cost, we outsource high computation load to cloud service providers without leaking file content and secret keys. Notably, our scheme can withstand collusion attack performed by revoked users cooperating with existing users. We prove the security of our scheme under the divisible computation Diffie-Hellman assumption. The result of our experiment shows computation cost for local devices is relatively low and can be constant. Our scheme is suitable for resource constrained devices.

**L5 :** **SeDaSC: Secure Data Sharing in Clouds**

**Authors :** Mazhar Ali, Revathi Dhamotharan, Eraj Khan, Samee U. Khan, Athanasios V. Vasilakos, Keqin Li, Albert Y. Zomaya**.**

**Published year : 2015**

**Description :** Cloud storage is an application of clouds that liberates organizations from establishing in-house data storage systems. However, cloud storage gives rise to security concerns. In case of group-shared data, the data face both cloud-specific and conventional insider threats. Secure data sharing among a group that counters insider threats of legitimate yet malicious users is an important research issue. In this paper, we propose the Secure Data Sharing in Clouds (SeDaSC) methodology that provides: 1) data confidentiality and integrity; 2) access control; 3) data sharing (forwarding) without using compute-intensive reencryption; 4) insider threat security; and 5) forward and backward access control. The SeDaSC methodology encrypts a file with a single encryption key. Two different key shares for each of the users are generated, with the user only getting one share. The possession of a single share of a key allows the SeDaSC methodology to counter the insider threats. The other key share is stored by a trusted third party, which is called the cryptographic server. The SeDaSC methodology is applicable to conventional and mobile cloud computing environments. We implement a working prototype of the SeDaSC methodology and evaluate its performance based on the time consumed during various operations. We formally verify the working of SeDaSC by using high-level Petri nets, the Satisfiability Modulo Theories Library, and a Z3 solver. The results proved to be encouraging and show that SeDaSC has the potential to be effectively used for secure data sharing in the cloud.

# L6 : IoT-Based Big Data Storage Systems in Cloud Computing: Perspectives and Challenges

**Authors :** Hongming Cai, Boyi Xu, Lihong Jiang, and Athanasios V. Vasilakos.

**Published year : 2016**

**Description :** Internet of Things (IoT) related applications have emerged as an important field for both engineers and researchers, reflecting the magnitude and impact of data-related problems to be solved in contemporary business organizations especially in cloud computing. This paper first provides a functional framework that identifies the acquisition, management, processing and mining areas of IoT big data, and several associated technical modules are defined and described in terms of their key characteristics and capabilities. Then current research in IoT application is analyzed, moreover, the challenges and opportunities associated with IoT big data research are identified. We also report a study of critical IoT application publications and research topics based on related academic and industry publications. Finally, some open issues and some typical examples are given under the proposed IoT-related research framework.

**L7:** **Enabling Semantic Search Based on Conceptual Graphs over Encrypted Outsourced Data**

**Authors :** Zhangjie Fu, Fengxiao Huang, Xingming Sun, Athanasios Vasilakos, and Ching-Nung Yang

**Published year: 2016**

**Description :** Currently, searchable encryption is a hot topic in the field of cloud computing. The existing achievements are mainly focused on keyword-based search schemes, and almost all of them depend on predefined keywords extracted in the phases of index construction and query. However, keyword-based search schemes ignore the semantic representation information of users' retrieval and cannot completely match users' search intention. Therefore, how to design a content-based search scheme and make semantic search more effective and context-aware is a difficult challenge. In this paper, for the first time, we define and solve the problems of semantic search based on conceptual graphs (CGs) over encrypted outsourced data in clouding computing (SSCG). We first employ the efficient measure of “sentence scoring” in text summarization and Tregex to extract the most important and simplified topic sentences from documents. We then convert these simplified sentences into CGs. To perform quantitative calculation of CGs, we design a new method that can map CGs to vectors. Next, we rank the returned results based on “text summarization score”. Furthermore, we propose a basic idea for SSCG and give a significantly improved scheme to satisfy the security guarantee of searchable symmetric encryption (SSE). Finally, we choose a real-world dataset, i.e., the CNN dataset to test our scheme. The results obtained from the experiment show the effectiveness of our proposed scheme.

**L8 :**  **Reducing Trust in the PKG in Identity Based Cryptosystems**

**Authors:** Vipul Goyal.

**Published year : 2007**

**Description :** One day, you suddenly find that a private key corresponding to your Identity is up for sale at e-Bay. Since you do not suspect a key compromise, perhaps it must be the PKG who is acting dishonestly and trying to make money by selling your key. How do you find out for sure and even prove it in a court of law? This paper introduces the concept of Traceable Identity based Encryption which is a new approach to mitigate the (inherent) key escrow problem in identity based encryption schemes. Our main goal is to restrict the ways in which the PKG can misbehave. In our system, if the PKG ever maliciously generates and distributes a decryption key for an Identity, it runs the risk of being caught and prosecuted. In contrast to other mitigation approaches, our approach does not require multiple key generation authorities.

**L9 :** **Traceable CP-ABE with Short Ciphertexts: How to Catch People Selling Decryption Devices on eBay Efficiently**

**Authors:** Jianting Ning, Zhenfu Cao, Xiaolei Dong,Junqing Gong Jie Chen

**Published year : 2016**

**Description :** Ciphertext-policy attribute-based encryption (CP-ABE) is a highly promising solution for cloud computing, which has been widely applied to provide fine-grained access control in cloud storage services recently. However, for CP-ABE based cloud storage systems, if a decryption device appears on eBay described and advertised to be able to decrypt any ciphertexts with policies satisfied by an attribute set or even with a specific access policy only, no one can trace the malicious user(s) who built such a decryption device using their private key(s). This has been known as a major obstacle to deploying CP-ABE systems in real-world commercial applications. Due to the one-to-many encryption mechanism of CP-ABE, the same decryption privilege is shared by multiple users who have the same attributes. It is difficult to identity the malicious user(s) who built such a decryption device. To track people selling decryption devices on eBay efficiently, in this paper, we develop a new methodology for constructing traitor tracing functionality, and present the first black-box traceable CP-ABE (BT-CP-ABE) with short ciphertexts which are independent of the number of users \(\mathcal {N}\). The black-box traceability is public, fully collusion-resistant, and adaptively traceable against both key-like decryption black-box and policy-specific decryption black-box

**SYSTEM REQUIREMENT**

H/W System Configuration:-

➢ Processor - Pentium –IV

➢ RAM - 4 GB (min)

➢ Hard Disk - 20 GB

➢ Key Board - Standard Windows Keyboard

➢ Mouse - Two or Three Button Mouse

➢ Monitor - SVGA

Software Requirements:

* Operating System - Windows XP
* Coding Language - Java/J2EE(JSP,Servlet)
* Front End - J2EE
* Back End - MySQL
* **Data Flow Diagram** :

Auditor

Public Cloud

Data User

System

Request

Response

Semi Trusted Authority

Data Owner

**Architecture Diagram**

View Users and Authorize

View Owners and Authorize

View Files

View File Transactions

View Top Searched Files

View Attackers

View Search Model

View Time Delay

View Throughput

**Data Owner**

`

Public Cloud

Attackers

Upload File

View Files

Send Trace Request and Trace Files

Delete Files

Transactions

Accepting all user Information

Req file and Res

**Data User**

Process all user queries

My Profile

View Files

Search Files

Search Ratio

Top K Search

Request Search Access Issue Credentials

**Auditor**

**Semi Trusted Authority**

Request Search Issue Credentials

View Files

View Trace Request and Give Permission

* **Flow Chart : Data Owner**

Register

Start

Login

Yes No

View Attackers

Username & Password Wrong

Logout

Upload Files

View All Your Files

Send Trace Request and Trace Files

Delete Files, Transactions

**Flow Chart : Data User**

**Start**

**User Login**

**Login**

Yes No

View My Profile

**Username & Password Wrong**

Logout

View Files

Search Files

Request Search Access Issue Credentials

* **Flow Chart : Public Cloud**

User Register

Start

Login

Yes No

View and Authorize Data Owners

View Data User and authorize

Username & Password Wrong

View All Files

View File Transactions

Logout

View All Attackers

View Search Model

View Time Delay

View Throughput

**Flow Chart: Auditor**

**Start**

**Login**

**Login**

Yes No

View Files

**Username & Password Wrong**

Logout

View Trace Request and Give Permission

**Flow Chart : Semi Trusted Authority**

Register

Start

Login

Yes No

Request Search Issue Credentials

Username & Password Wrong

Logout

**Class Diagram :**

**Public Cloud**

**Auditor**

View Users and Authorize, View Owners and Authorize ,View Files, View File Transactions, View Top Searched Files, View Attackers , View Search Model, View Time Delay, View Throughput

Select File, Index name, File Name, Date & Time, Data Owner, Trapdoor-MAC, rank

View Files, View Trace Request and Give Permission

Select File, Index name, File Name, Date & Time, Data Owner, Trapdoor-MAC, rank

Methods

Methods

Members

Members

**Register**

**Login**

Login (), Reset (), Register ().

User Name, Password.

Register (), Reset ()

User Name, Password, E-mail, Mobile, Address, DOB, Gender, Pin code, Image

Login, Register

User Name, Password

Methods

Methods

Members

Members

**Data User**

Methods

View my Profile, View Files , Search Files, Search Ratio, Top K Search, Request Search Access Issue Credentials

Select File, Index name, File Name, Date & Time, Data Owner, Trapdoor-MAC, rank

**Semi Trusted Authority**

**Data Owner**

cbbvbvcvv

Attackers, Upload File, View Files, Send Trace Request and Trace Files, Delete Files, Transactions

Select File, Index name, File Name, Date & Time, Data Owner, Trapdoor-MAC, rank

Request Search Issue Credentials

Select File, Index name, File Name, Date & Time, Data Owner, Trapdoor-MAC, rank

Members

* **Sequence Diagram**

Public Cloud

Auditor

Semi Trusted Authority

Data Owner

Data User

View data users and authorize

Register and Login

Request Search Issue Credentials

View and Authorize Data Owner

**Data Analizer**

Upload and view your files

View Files

View files

Request Search Access Issue Credentials

View file transactions

Send Trace Request and Trace Files

View Attackers

View Trace Request and Give Permission

View Time Delay

Top K search

**IMPLEMENTATION**

**Data Owner**

In this module, the data owner performs operations such as Attackers, Upload File, View Files, Send Trace Request and Trace Files, Delete Files, Transactions

**Data User**

In this module, he logs in by using his/her user name and password. After Login receiver will perform operations like View my Profile, View Files , Search Files, Search Ratio, Top K Search, Request Search Access Issue Credentials

**Auditor**

In this module, the sector can do following operations View Files, View Trace Request and Give Permission

**Semi Trusted Authority**

In this module, the sector can do following operations Request Search Issue Credentials

**Public Cloud**

The Cloud manages a server to provide data storage service and can also do the following operations such as View Users and Authorize, View Owners and Authorize ,View Files, View File Transactions, View Top Searched Files, View Attackers , View Search Model, View Time Delay, View Throughput

**SYSTEM TESTING**

### TESTING METHODOLOGIES

The following are the Testing Methodologies:

* **Unit Testing.**
* **Integration Testing.**
* **User Acceptance Testing.**
* **Output Testing.**
* **Validation Testing.**

**Unit Testing**

Unit testing focuses verification effort on the smallest unit of Software design that is the module. Unit testing exercises specific paths in a module’s control structure to

ensure complete coverage and maximum error detection. This test focuses on each module individually, ensuring that it functions properly as a unit. Hence, the naming is Unit Testing.

During this testing, each module is tested individually and the module interfaces are verified for the consistency with design specification. All important processing path are tested for the expected results. All error handling paths are also tested.

**Integration Testing**

Integration testing addresses the issues associated with the dual problems of verification and program construction. After the software has been integrated a set of high order tests are conducted. The main objective in this testing process is to take unit tested modules and builds a program structure that has been dictated by design.

**The following are the types of Integration Testing:**

1. **Top Down Integration**

This method is an incremental approach to the construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main program module. The module subordinates to the main program module are incorporated into the structure in either a depth first or breadth first manner.

In this method, the software is tested from main module and individual stubs are replaced when the test proceeds downwards.

**2. Bottom-up Integration**

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs is eliminated. The bottom up integration strategy may be implemented with the following steps:

* The low-level modules are combined into clusters into clusters that perform a specific Software sub-function.
* A driver (i.e.) the control program for testing is written to coordinate test case input and output.
* The cluster is tested.
* Drivers are removed and clusters are combined moving upward in the program structure

The bottom up approaches tests each module individually and then each module is module is integrated with a main module and tested for functionality.

**User Acceptance Testing**

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

**Output Testing**

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways – one is on screen and another in printed format.

**Validation Checking**

Validation checks are performed on the following fields.

**Text Field:**

The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables. Incorrect entry always flashes and error message.

**Numeric Field:**

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error messages. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system. Testing involves executing the real data information is used in the program the existence of any program defect is inferred from the output. The testing should be planned so that all the requirements are individually tested.

A successful test is one that gives out the defects for the inappropriate data and produces and output revealing the errors in the system.

**Preparation of Test Data**

Taking various kinds of test data does the above testing. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

**Using Live Test Data:**

Live test data are those that are actually extracted from organization files. After a system is partially constructed, programmers or analysts often ask users to key in a set of data from their normal activities. Then, the systems person uses this data as a way to partially test the system. In other instances, programmers or analysts extract a set of live data from the files and have them entered themselves.

It is difficult to obtain live data in sufficient amounts to conduct extensive testing. And, although it is realistic data that will show how the system will perform for the typical processing requirement, assuming that the live data entered are in fact typical, such data generally will not test all combinations or formats that can enter the system. This bias toward typical values then does not provide a true systems test and in fact ignores the cases most likely to cause system failure.

**Using Artificial Test Data:**

Artificial test data are created solely for test purposes, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information systems department, make possible the testing of all login and control paths through the program.

The most effective test programs use artificial test data generated by persons other than those who wrote the programs. Often, an independent team of testers formulates a testing plan, using the systems specifications.

The package “Virtual Private Network” has satisfied all the requirements specified as per software requirement specification and was accepted.

**USER TRAINING**

Whenever a new system is developed, user training is required to educate them about the working of the system so that it can be put to efficient use by those for whom the system has been primarily designed. For this purpose the normal working of the project was demonstrated to the prospective users. Its working is easily understandable and since the expected users are people who have good knowledge of computers, the use of this system is very easy.

**MAINTAINENCE**

This covers a wide range of activities including correcting code and design errors. To reduce the need for maintenance in the long run, we have more accurately defined the user’s requirements during the process of system development. Depending on the requirements, this system has been developed to satisfy the needs to the largest possible extent. With development in technology, it may be possible to add many more features based on the requirements in future. The coding and designing is simple and easy to understand which will make maintenance easier.

**TESTING STRATEGY :**

A strategy for system testing integrates system test cases and design techniques into a well planned series of steps that results in the successful construction of software. The testing strategy must co-operate test planning, test case design, test execution, and the resultant data collection and evaluation .A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions against user requirements.

Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing represents an interesting anomaly for the software. Thus, a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

**SYSTEM TESTING:**

Software once validated must be combined with other system elements (e.g. Hardware, people, database). System testing verifies that all the elements are proper and that overall system function performance is

achieved. It also tests to find discrepancies between the system and its original objective, current specifications and system documentation.

**UNIT TESTING:**

In unit testing different are modules are tested against the specifications produced during the design for the modules. Unit testing is essential for verification of the code produced during the coding phase, and hence the goals to test the internal logic of the modules. Using the detailed design description as a guide, important Conrail paths are tested to uncover errors within the boundary of the modules. This testing is carried out during the programming stage itself. In this type of testing step, each module was found to be working satisfactorily as regards to the expected output from the module.

In Due Course, latest technology advancements will be taken into consideration. As part of technical build-up many components of the networking system will be generic in nature so that future projects can either use or interact with this.The future holds a lot to offer to the development and refinement of this project.

**PRELIMINARY INVESTIGATION**

The first and foremost strategy for development of a project starts from the thought of designing a mail enabled platform for a small firm in which it is easy and convenient of sending and receiving messages, there is a search engine ,address book and also including some entertaining games. When it is approved by the organization and our project guide the first activity, ie. preliminary investigation begins. The activity has three parts:

* **Request Clarification**
* **Feasibility Study**
* **Request Approval**

**REQUEST CLARIFICATION**

After the approval of the request to the organization and project guide, with an investigation being considered, the project request must be examined to determine precisely what the system requires.

Here our project is basically meant for users within the company whose systems can be interconnected by the Local Area Network(LAN). In today’s busy schedule man need everything should be provided in a readymade manner. So taking into consideration of the vastly use of the net in day to day life, the corresponding development of the portal came into existence.

**FEASIBILITY ANALYSIS**

An important outcome of preliminary investigation is the determination that the system request is feasible. This is possible only if it is feasible within limited resource and time. The different feasibilities that have to be analyzed are

* **Operational Feasibility**
* **Economic Feasibility**
* **Technical Feasibility**

###### Operational Feasibility

Operational Feasibility deals with the study of prospects of the system to be developed. This system operationally eliminates all the tensions of the Admin and helps him in effectively tracking the project progress. This kind of automation will surely reduce the time and energy, which previously consumed in manual work. Based on the study, the system is proved to be operationally feasible.

**Economic Feasibility**

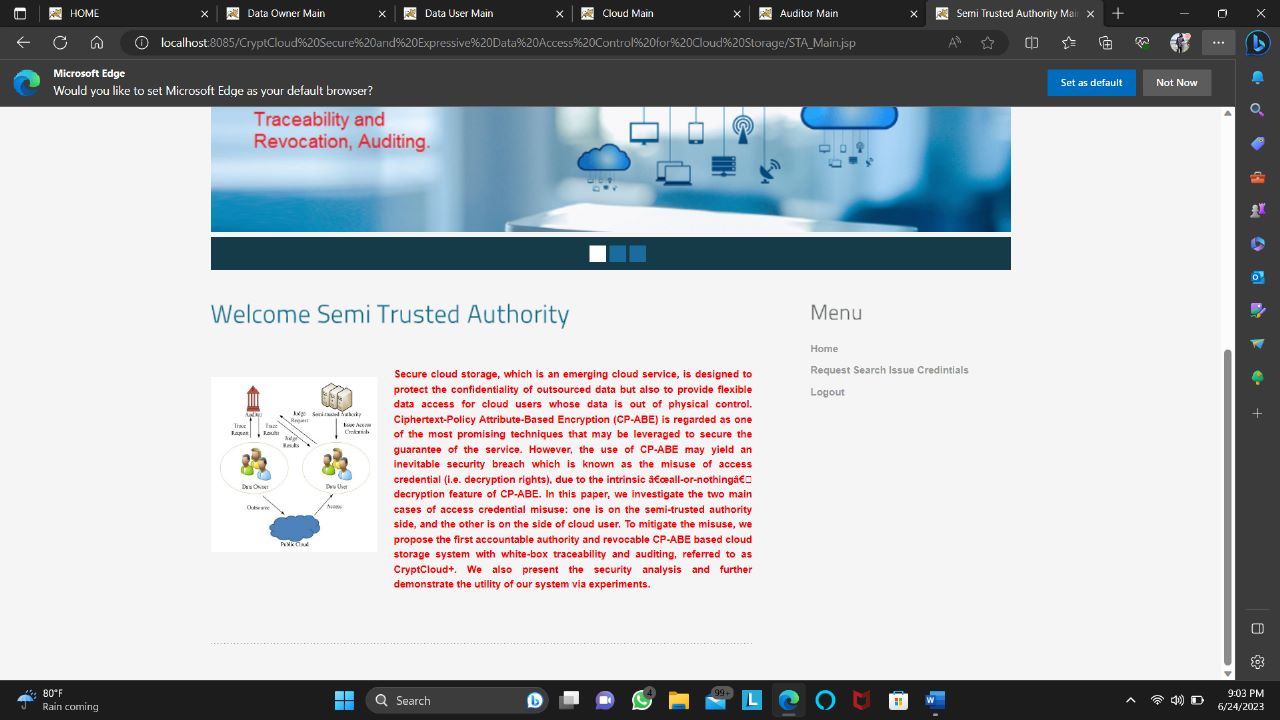
Economic Feasibility or Cost-benefit is an assessment of the economic justification for a computer based project. As hardware was installed from the beginning & for lots of purposes thus the cost on project of hardware is low. Since the system is a network based, any number of employees connected to the LAN within that organization can use this tool from at anytime. The Virtual Private Network is to be developed using the existing resources of the organization. So the project is economically feasible.

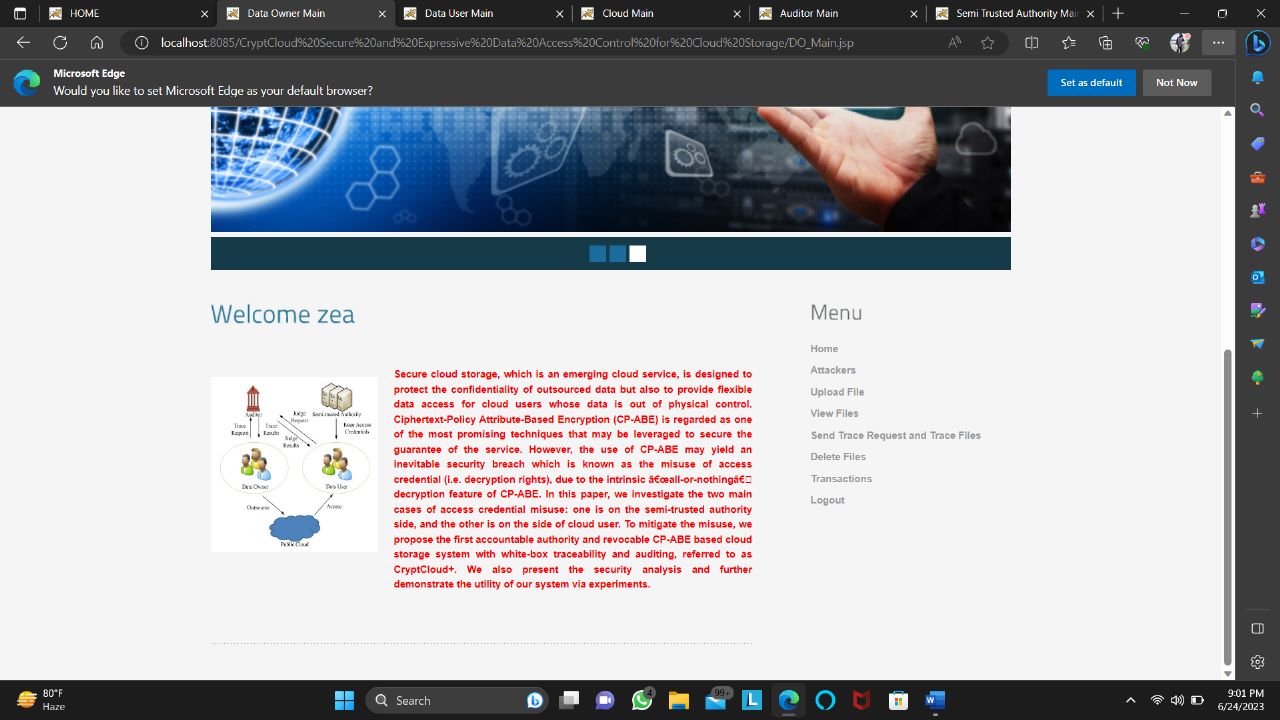
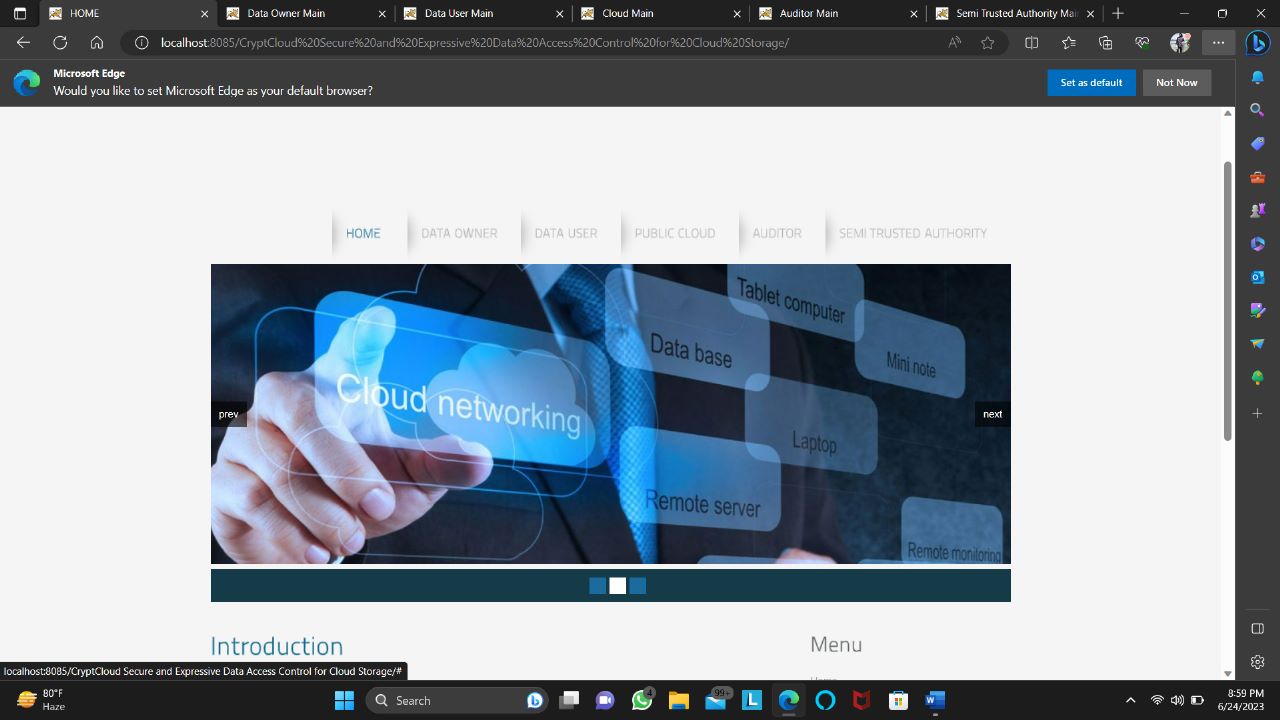
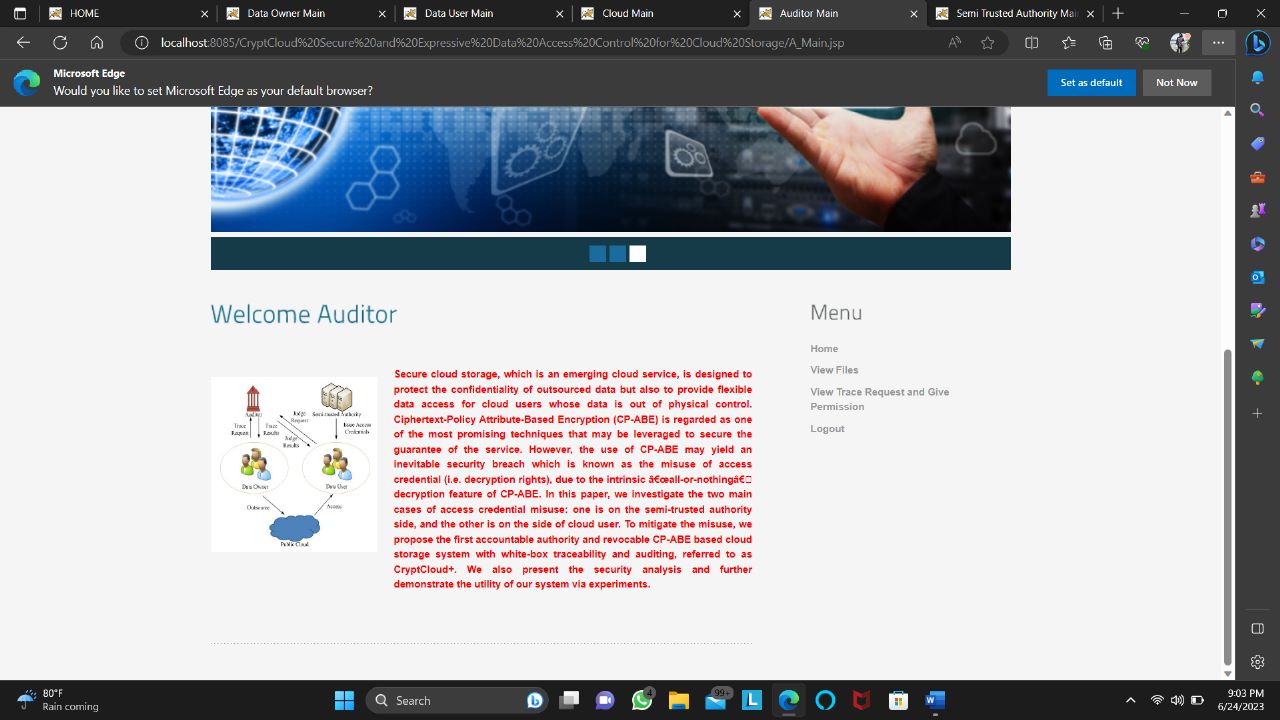
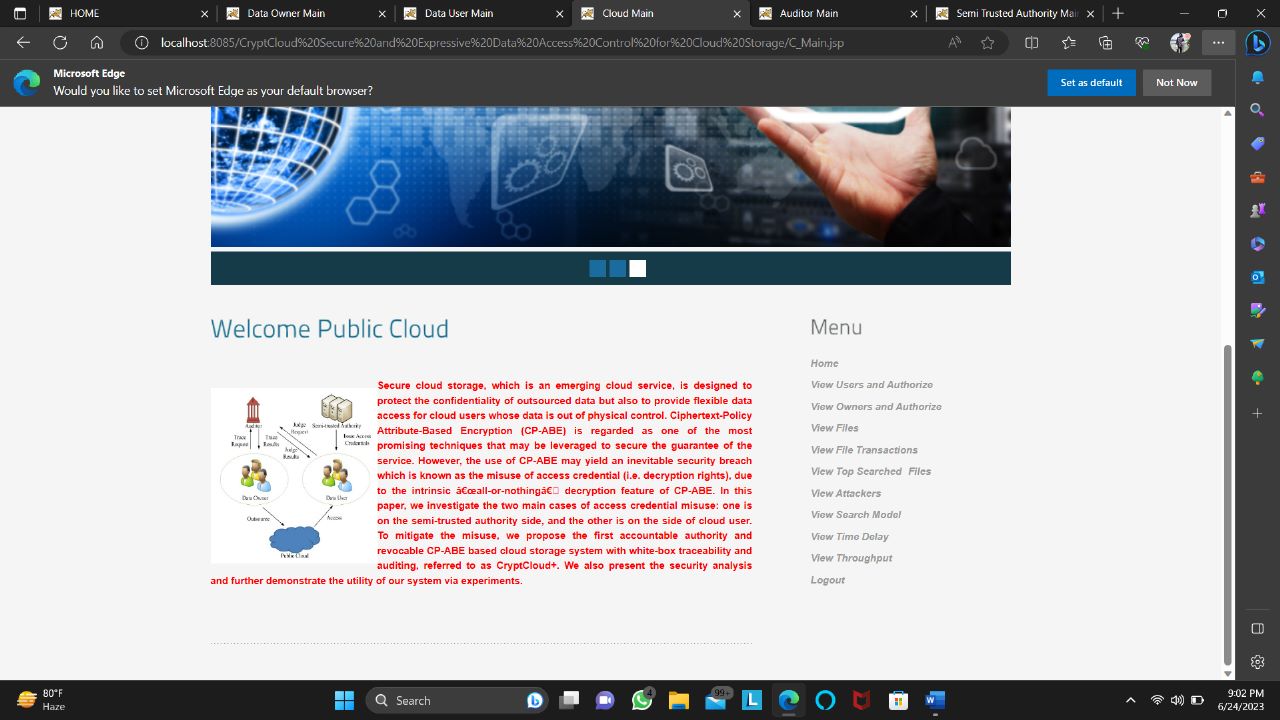
###### Technical Feasibility

According to Roger S. Pressman, Technical Feasibility is the assessment of the technical resources of the organization. The organization needs IBM compatible machines with a graphical web browser connected to the Internet and Intranet. The system is developed for platform Independent environment. Java Server Pages, JavaScript, HTML, SQL server and WebLogic Server are used to develop the system. The technical feasibility has been carried out. The system is technically feasible for development and can be developed with the existing facility.

**REQUEST APPROVAL**

Not all request projects are desirable or feasible. Some organization receives so many project requests from client users that only few of them are pursued. However, those projects that are both feasible and desirable should be put into schedule. After a project request is approved, it cost, priority, completion time and personnel requirement is estimated and used to determine where to add it to any project list. Truly speaking, the approval of those above factors, development works can be launched.

****

****

**SYSTEM DESIGN AND DEVELOPMENT**

**INPUT DESIGN**

Input Design plays a vital role in the life cycle of software development, it requires very careful attention of developers. The input design is to feed data to the application as accurate as possible. So inputs are supposed to be designed effectively so that the errors occurring while feeding are minimized. According to Software Engineering Concepts, the input forms or screens are designed to provide to have a validation control over the input limit, range and other related validations.

This system has input screens in almost all the modules. Error messages are developed to alert the user whenever he commits some mistakes and guides him in the right way so that invalid entries are not made. Let us see deeply about this under module design.

Input design is the process of converting the user created input into a computer-based format. The goal of the input design is to make the data entry logical and free from errors. The error is in the input are controlled by the input design. The application has been developed in user-friendly manner. The forms have been designed in such a way during the processing the cursor is placed in the position where must be entered. The user is also provided with in an option to select an appropriate input from various alternatives related to the field in certain cases.

Validations are required for each data entered. Whenever a user enters an erroneous data, error message is displayed and the user can move on to the subsequent pages after completing all the entries in the current page.

OUTPUT DESIGN

The Output from the computer is required to mainly create an efficient method of communication within the company primarily among the project leader and his team members, in other words, the administrator and the clients. The output of VPN is the system which allows the project leader to manage his clients in terms of creating new clients and assigning new projects to them, maintaining a record of the project validity and providing folder level access to each client on the user side depending on the projects allotted to him. After completion of a project, a new project may be assigned to the client. User authentication procedures are maintained at the initial stages itself. A new user may be created by the administrator himself or a user can himself register as a new user but the task of assigning projects and validating a new user rests with the administrator only.

The application starts running when it is executed for the first time. The server has to be started and then the internet explorer in used as the browser. The project will run on the local area network so the server machine will serve as the administrator while the other connected systems can act as the clients. The developed system is highly user friendly and can be easily understood by anyone using it even for the first time.

**CONCLUSION AND FUTURE WORK**

In this work, we have addressed the challenge of credential leakageinCP-ABE based cloud storage system by designing an accountable authority and revocable CryptCloud which supports white-box traceability and auditing (referred to as CryptCloud+). This is the ﬁrst CP-ABE based cloud storage system that simultaneously supports white-box traceability, accountable authority, auditing and effective revocation. Speciﬁcally, CryptCloud+ allows us to trace and revoke malicious cloud users (leaking credentials). Our approach can be also used in the case where the users’ credentials are redistributed by the semi-trusted authority. We note that we may need black-box traceability, which is a stronger notion (compared to white-box traceability), in CryptCloud. One of our future works is to consider the black-box traceability and auditing. Furthermore, AU is assumed to be fully trusted in CryptCloud+. However, in practice, it may not be the case. Is there any way to reduce trust from AU? Intuitively, one method is to employ multiple AUs. This is similar to the technique used in threshold schemes. But it will require additional communication andd eploymen tcostand meanwhile, the problem of collusion among AUs remains. Another potential approach is to employ secure multi-party computation in the presence of malicious adversaries. However, the efﬁciency is also a bottleneck. Designing efﬁcient multi-party computation and decentralizing trust among AUs (while maintaining the same level of security and efﬁciency) is also a part of our future work. We use Paillier-like encryption to serve as an extractable commitment to achieve white-box traceability. From an abstract view point, any extractable commitment may be

employed to achieve white-box traceability in theory. To improve the efﬁciency of tracing, we may make use of a more light-weight (pairing-suitable) extractable commitment. Also, the trace algorithm in CryptCloud+ needs to take the master secret key as input to achieve white-box traceability of malicious cloud users. Intuitively, the proposed CryptCloud+ is private traceable5. Private traceability only allows the tracing algorithm to be run by the system administrator itself, while partial/full public traceability enables the administrator, authorized users and even anyone without the secret information of the system to fulﬁll the trace. Our future work will include extending CryptCloud+ to provide “partial” and fully public traceability without compromising on performance.

**BIBLIOGRAPHY**

[1] Mazhar Ali, Revathi Dhamotharan, Eraj Khan, Samee U. Khan, Athanasios V. Vasilakos, Keqin Li, and Albert Y. Zomaya. Sedasc: Secure data sharing in clouds. IEEE Systems Journal, 11(2):395–404, 2017.

[2] MazharAli,SameeU.Khan,andAthanasiosV.Vasilakos. Security in cloud computing: Opportunities and challenges. Inf. Sci., 305:357–383, 2015.

[3] Michael Armbrust, Armando Fox, Rean Grifﬁth, Anthony D Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, et al. A view of cloud computing. Communications of the ACM, 53(4):50–58, 2010.

[4] Nuttapong Attrapadung and Hideki Imai. Attribute-based encryption supporting direct/indirect revocation modes. In Cryptography and Coding, pages 278–300. Springer, 2009.

[5] Amos Beimel. Secure schemes for secret sharing and key distribution. PhD thesis, PhD thesis, Israel Institute of Technology, Technion, Haifa, Israel, 1996.

[6] Mihir Bellare and Oded Goldreich. On deﬁning proofs of knowledge. In Advances in Cryptology-CRYPTO’92, pages 390–420. Springer, 1993.

[7] Dan Boneh and Xavier Boyen. Short signatures without random oracles. In EUROCRYPT - 2004, pages 56–73, 2004.

[8] Hongming Cai, Boyi Xu, Lihong Jiang, and Athanasios V. Vasilakos. Iot-based big data storage systems in cloud computing: Perspectives and challenges. IEEE Internet of Things Journal, 4(1):75–87, 2017.

[9] Jie Chen, Romain Gay, and Hoeteck Wee. Improved dual system ABE in prime-order groups via predicate encodings. In Advances in Cryptology - EUROCRYPT 2015, pages 595–624, 2015.

[10] Angelo De Caro and Vincenzo Iovino. jpbc: Java pairing based cryptography. In ISCC 2011, pages 850–855. IEEE, 2011.

[11] Hua Deng, Qianhong Wu, Bo Qin, Jian Mao, Xiao Liu, Lei Zhang, and Wenchang Shi. Who is touching my cloud. In Computer Security-ESORICS 2014, pages 362–379. Springer, 2014.

[12] Zhangjie Fu, Fengxiao Huang, Xingming Sun, Athanasios Vasilakos, and Ching-Nung Yang. Enabling semantic search based on conceptual graphs over encrypted outsourced data. IEEE Transactions on Services Computing, 2016.

5. As noted in [36], there three types of traceability, namely: private traceability, partial public traceability and fully public traceability.

1939-1374 (c) 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See <http://www.ieee.org/publications_standards/publications/rights/index.html> for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2791538, IEEE Transactions on Services Computing

13

[13] Vipul Goyal. Reducing trust in the PKG in identity based cryptosystems. InAdvancesinCryptology-CRYPTO2007,pages430–447. Springer, 2007.

[14] Vipul Goyal, Steve Lu, Amit Sahai, and Brent Waters. Black-box accountable authority identity-based encryption. In Proceedings of the 15th ACM conference on Computer and communications security, pages 427–436. ACM, 2008.

[15] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for ﬁne-grained access control of encrypted data. InProceedingsofthe13thACMconferenceonComputer and communications security, pages 89–98. ACM, 2006.

[16] Qi Jing, Athanasios V. Vasilakos, Jiafu Wan, Jingwei Lu, and Dechao Qiu. Security of the internet of things: perspectives and challenges. Wireless Networks, 20(8):2481–2501, 2014.

[17] Allison Lewko. Tools for simulating features of composite order bilinear groups in the prime order setting. In Advances in Cryptology–EUROCRYPT 2012, pages 318–335. Springer, 2012.

[18] Allison Lewko, Tatsuaki Okamoto, Amit Sahai, Katsuyuki Takashima, and Brent Waters. Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In Advances in Cryptology–EUROCRYPT 2010, pages 62–91. Springer, 2010.

[19] Allison Lewko and Brent Waters. New proof methods for attribute-based encryption: Achieving full security through selective techniques. In Advances in Cryptology–CRYPTO 2012, pages 180–198. Springer, 2012. [20] Jiguo Li, Xiaonan Lin, Yichen Zhang, and Jinguang Han. KSFOABE: outsourced attribute-based encryption with keyword search function for cloud storage. IEEE Trans. Services Computing, 10(5):715–725, 2017.

[21] JiguoLi,WeiYao,YichenZhang,HuilingQian,andJinguangHan. Flexible and ﬁne-grained attribute-based data storage in cloud computing. IEEE Trans. Services Computing, 10(5):785–796, 2017.

[22] Jin Li, Qiong Huang, Xiaofeng Chen, Sherman SM Chow, Duncan S Wong, and Dongqing Xie. Multi-authority ciphertext-policy attribute-basedencryptionwithaccountability. InProceedingsofthe 6th ACM Symposium on Information, Computer and Communications Security, ASIACCS 2011, pages 386–390. ACM, 2011.

[23] Jin Li, Kui Ren, and Kwangjo Kim. A2be: Accountable attributebased encryption for abuse free access control. IACR Cryptology ePrint Archive, 2009:118, 2009.

[24] Jiaqiang Liu, Yong Li, Huandong Wang, Depeng Jin, Li Su, Lieguang Zeng, and Thanos Vasilakos. Leveraging softwaredeﬁned networking for security policy enforcement. Inf. Sci., 327:288–299, 2016.

[25] QiangLiu,HaoZhang,JiafuWan,andXinChen. Anaccesscontrol model for resource sharing based on the role-based access control intendedformulti-domainmanufacturinginternetofthings. IEEE Access, 5:7001–7011, 2017.

[26] Zhen Liu, Zhenfu Cao, and Duncan S Wong. Blackbox traceable cp-abe: how to catch people leaking their keys by selling decryption devices on ebay. In Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security, pages 475–486. ACM, 2013.